



# **PREMIUM AUDIT**

NAME OF PROJECT	CalorieToken
TOKEN BEP20 ADDRESS	0x100bDE487795d4539bAc6981E126a70A8DfBCf1E
WEBSITE / TELEGRAM	<a href="https://calorie.finance/">https://calorie.finance/</a>



**CalorieToken**

**DISCLAIMER: PLEASE READ FULL AUDIT**



# IMPORTANT DISCLAIMER

---

I.

## **NOT RESPONSIBLE**

Analytix Audit holds no responsibility for any actions from the project in this audit.

II.

## **NOT GUARANTEE**

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

III.

## **INFORMATION**

Analytix Audit researches and provides public information about the project in an easy-to-understand format for the common person.

IV.

## **AUDIT AGREEMENT**

This audit agreement does not guarantee ANY illicit actions by the project team and does not serve as an advocacy for the project.

V.

## **NO ADVICE**

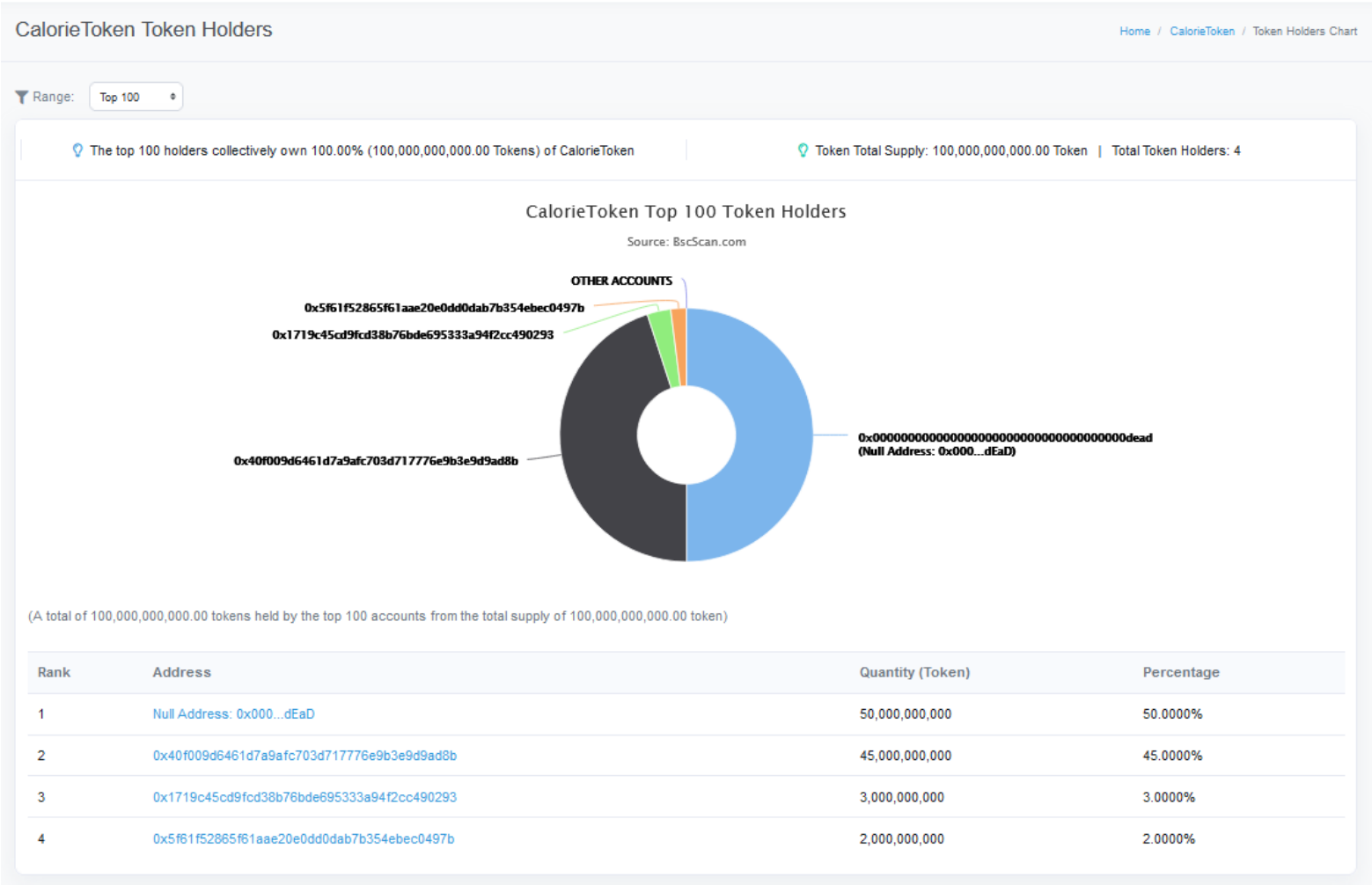
Analytix Audit in no way takes responsibility for any losses, not does Analytix Audit encourage any speculative investments.

VI.

## **DISCLAIMER**

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

# TOKENOMICS - OVERVIEW



- More than 15% of the tokens are unlocked

(This comment refers to when the audit was carried out maybe team locked tokens later)

# CONTRACT CODE - OVERVIEW



## QUICK REVIEW

- Owner can change fees up to 100%
  - Owner can't set Tx Limits.
- Owner can blacklist multiple wallets in a single transaction

# Owner Privileges:

```
520 * Requirements:
521 *
522 * - the calling contract must have an ETH balance of at least `value`.
523 * - the called Solidity function must be `payable`.
524 *
525 * _Available since v3.1._
526 */
527 function functionCallWithValue(
528     address target,
529     bytes memory data,
530     uint256 value
531 ) internal returns (bytes memory) {
532     return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533 }
534
535 /**
536 * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
537 * with `errorMessage` as a fallback revert reason when `target` reverts.
538 *
539 * _Available since v3.1._
540 */
541 function functionCallWithValue(
542     address target,
543     bytes memory data,
544     uint256 value,
545     string memory errorMessage
546 ) internal returns (bytes memory) {
547     require(address(this).balance >= value, "Address: insufficient balance for call");
548     require(isContract(target), "Address: call to non-contract");
549
550     (bool success, bytes memory returndata) = target.call(value: value)(data);
```

```
function waiveOwnership() public virtual onlyOwner {
function transferOwnership(address newOwner) public virtual onlyOwner {
function setMarketPairStatus(address account, bool newValue) public onlyOwner {
function setIsExcludedFromFee(address account, bool newValue) public onlyOwner {
function setBuyTaxes(uint256 newLiquidityTax, uint256 newMarketingTax, uint256 newAppTax, uint256
    newBuybackTax) external onlyOwner() {
function setSellTaxes(uint256 newLiquidityTax, uint256 newMarketingTax, uint256 newAppTax, uint256
    newBuybackTax) external onlyOwner() {
function blacklistAddress(address account, bool value) external onlyOwner{
    (Owner can blacklist wallet)
function blacklistMultipleAddress(address[] calldata accounts, bool value) public onlyOwner {
function setDistributionSettings(uint256 newLiquidityShare, uint256 newMarketingShare, uint256
    newAppShare, uint256 newBuybackShare) external onlyOwner() {
function setNumTokensBeforeSwap(uint256 newLimit) external onlyOwner() {
function setMarketingWalletAddress(address newAddress) external onlyOwner() {
function setAppWalletAddress(address newAddress) external onlyOwner() {
function setBuybackWalletAddress(address newAddress) external onlyOwner() {
function setSwapAndLiquifyEnabled(bool _enabled) public onlyOwner {
function setSwapAndLiquifyByLimitOnly(bool newValue) public onlyOwner {
function changeRouterVersion(address newRouterAddress) public onlyOwner returns(address
    newPairAddress) {
```

# Issue Checking:

N°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function security.	Passed

Audit Result:

**PASSED**

# WEBSITE - OVERVIEW



🕒 **Domain Registration: N/A**

N/A day old

**SSL CERTIFICATE: N/A**

(<https://www.ssllabs.com>)

# SOCIAL MEDIA - OVERVIEW



CALORIETOKENGLOBAL



CALORIETOKENBSC



CALORIE.FINANCE